Computer Components

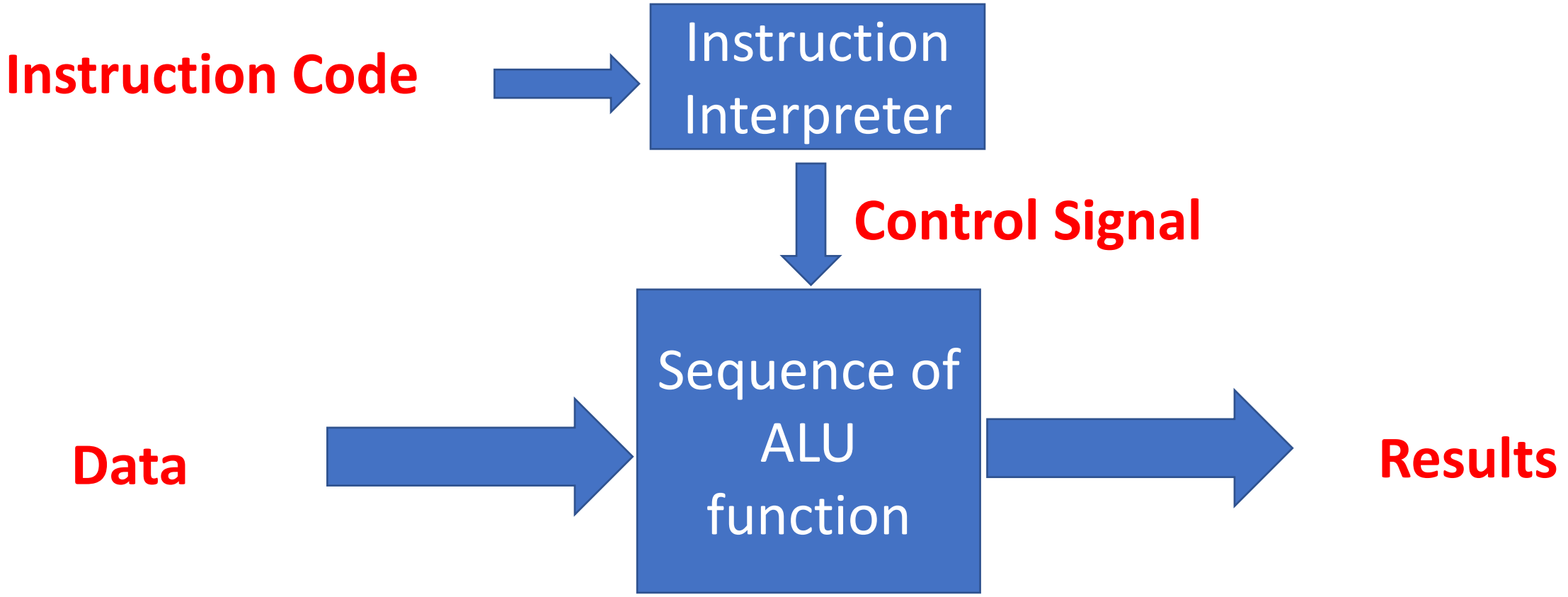
CSS 122

From von Neumann architecture

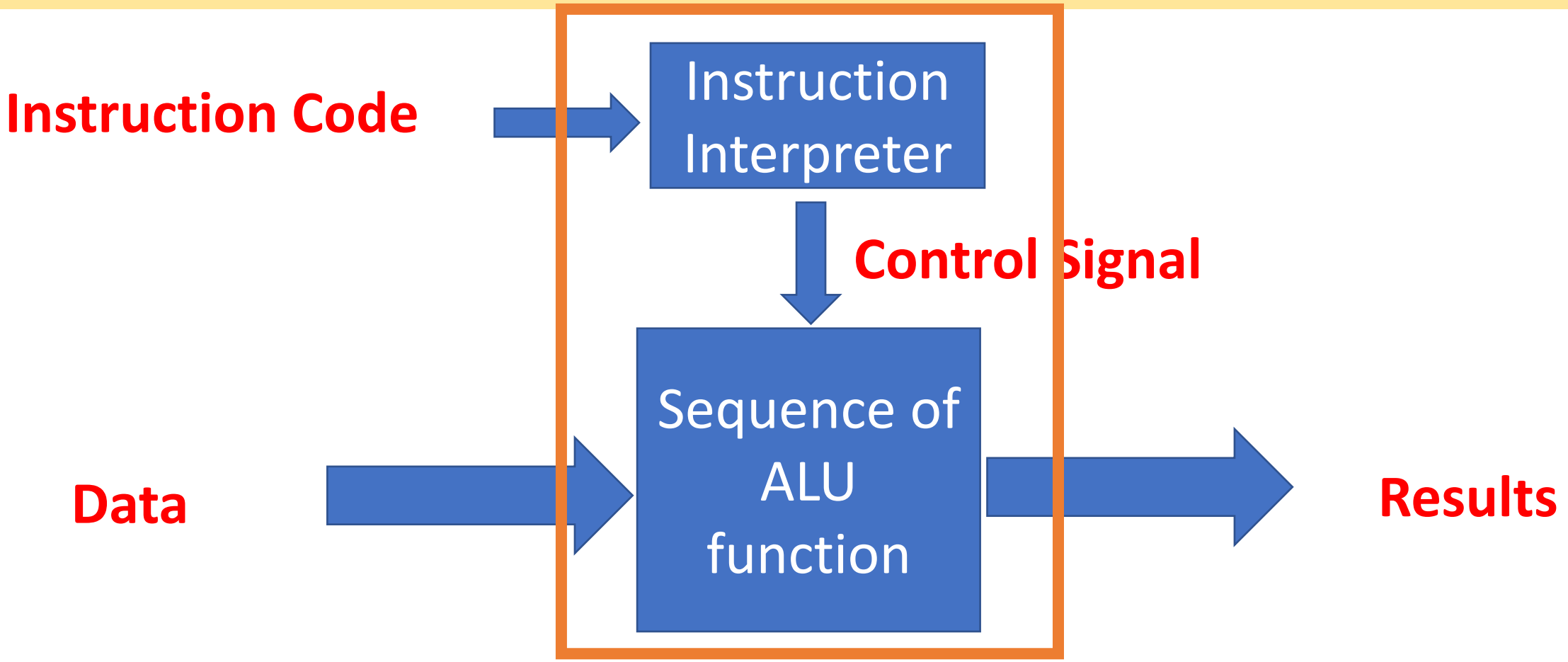
- Data and instruction are stored in a single read-write memory
- The contents in the memory are addressed by location without regard to the type of data contained there
- Execution occur in sequential fashion unless explicitly modified from on instruction to the next.

Hardware Program

Hardware and Software Approach



The CPU

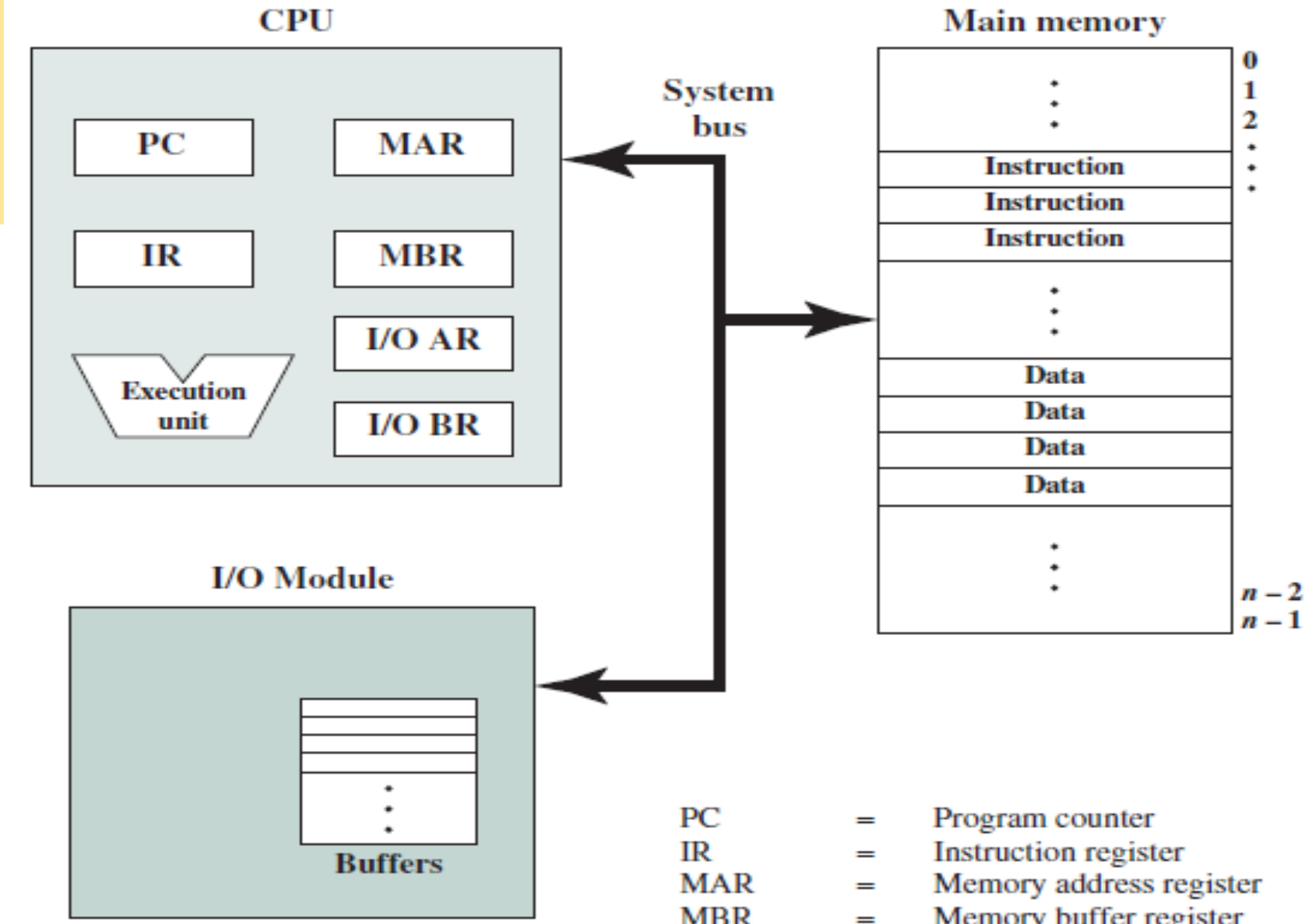

Instruction Code
Instruction Interpreter
Control Signal
Sequence of ALU function
Data
Results

Data, Instruction and Results

- Data and instructions must be put into the system
- In order to Data and instructions to the system we need some sort of input module
- This module contains basic components for accepting data and instructions in some form and converting them into an internal form of signals usable by the system.
- A means of reporting results is needed, and this is in the form of an output module
- Together, these are referred to as I/O components

Main memory

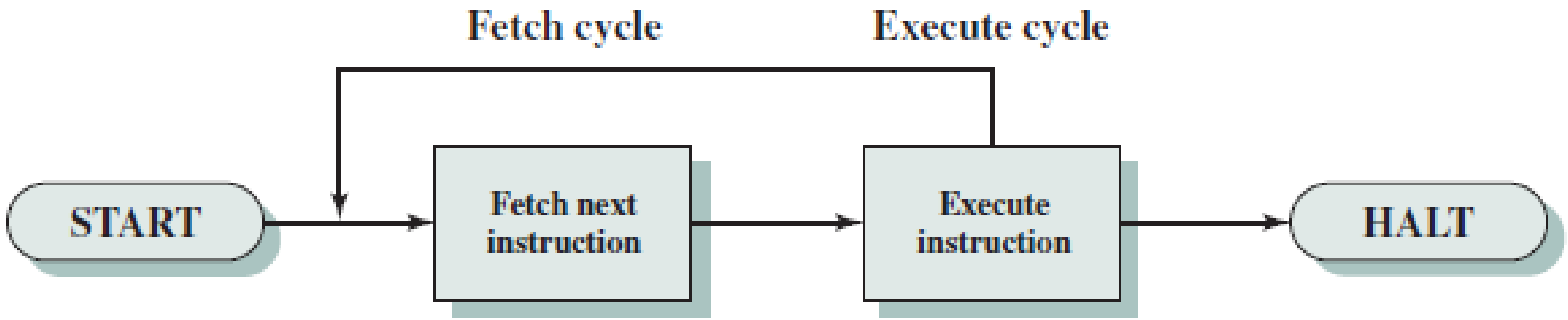
- An input device bring instructions and data in sequentially
- But a program is not invariably executed sequentially it may jump around randomly
- Operations on data may require access to more than just one element at a time in a predetermined sequence
- There must be a place to temporarily store both instructions and data
- That module is called memory, or main memory


CPU
PC
MAR
IR
MBR
I/O AR
Execution unit
I/O BR
System bus
Main memory
0
1
2
Instruction
Instruction
Instruction
Data
Data
Data
Data
n – 2
n – 1
I/O Module
Buffers
PC = Program counter
IR = Instruction register
MAR = Memory address register
MBR = Memory buffer register
I/O AR = Input/output address register
I/O BR = Input/output buffer register

Computer Function

- The basic function performed by a computer is execution of a program, which consists of a set of instructions stored in memory
- Instruction processing consists of two steps:
  - ❖The processor reads (fetches) instructions
  - ❖The processor executes each instruction
- The processing required for a single instruction is called an Instruction Cycle

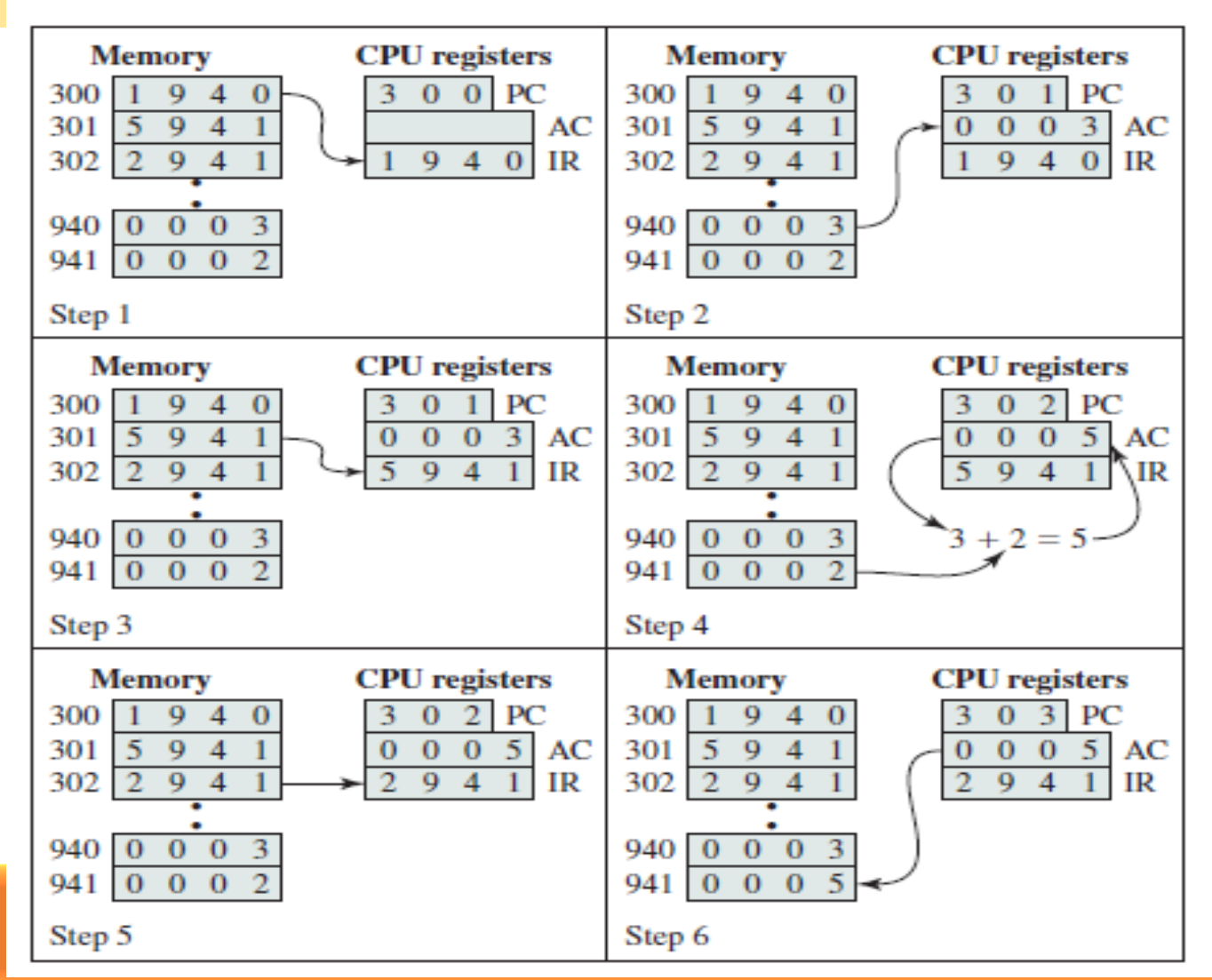
Instruction Fetch and Execute



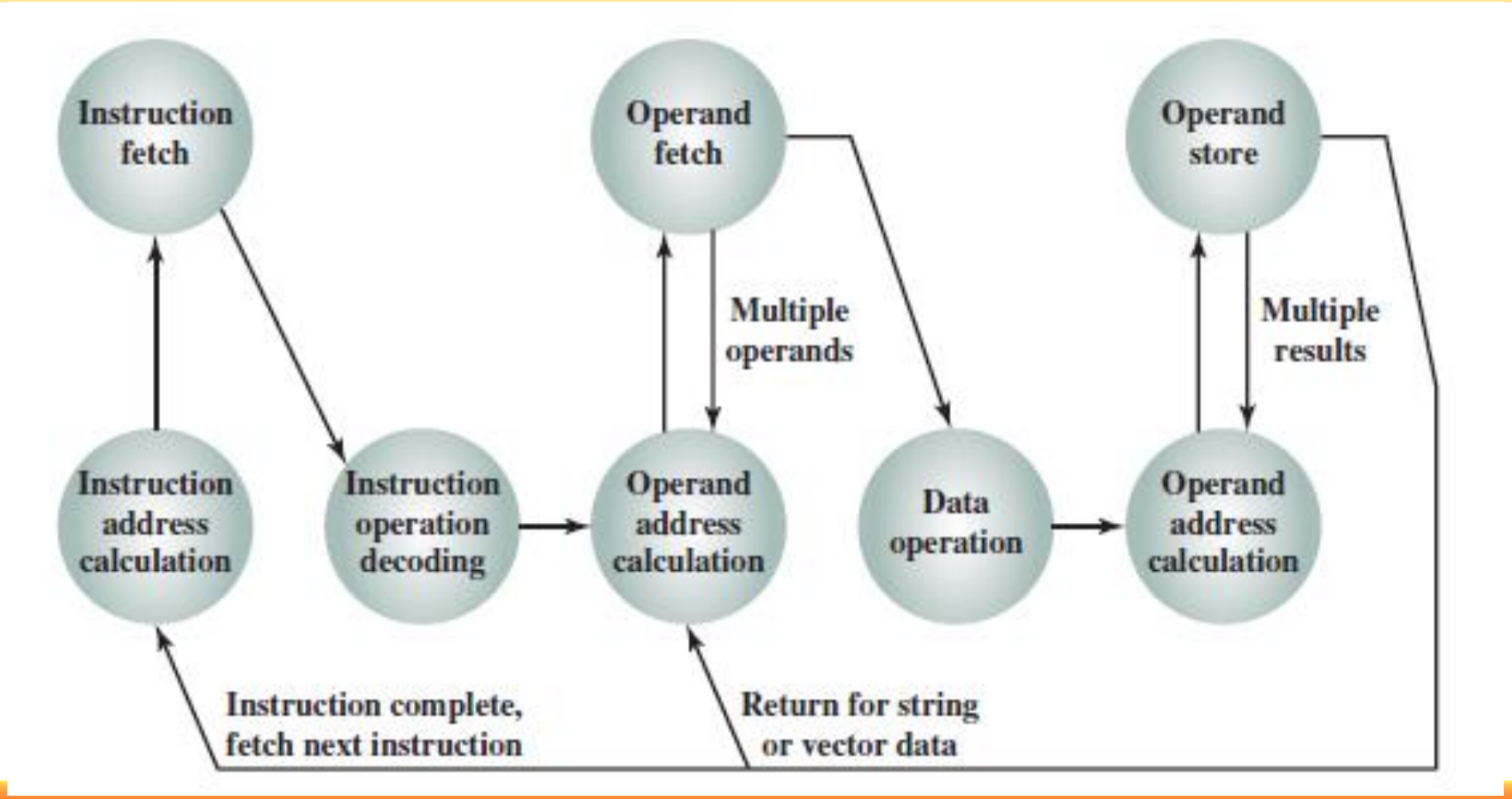
Category of Action Performed by Processor

- Processor-memory: Data may be transferred from processor to memory or from memory to processor.
- Processor-I/O: Data may be transferred to or from a peripheral device by transferring between the processor and an I/O module.
- Data processing: The processor may perform some arithmetic or logic operation on data.
- Control: An instruction may specify that the sequence of execution be altered.

Example of Program in Binary/Hexadecimal



Instruction Cycle State Diagram



Fetch Cycle operation

- **Instruction fetch (if):** Read instruction from its memory location into the processor
- **Instruction operation decoding (iod):** Analyze instruction to determine type of operation to be performed and operand(s) to be used.
- **Operand address calculation (oac):** If the operation involves reference to an operand in memory or available via I/O, then determine the address of the operand
- **Operand fetch (of):** Fetch the operand from memory or read it in from I/O.

Fetch Cycle operation Cont.

- **Data operation (do):** Perform the operation indicated in the instruction.
- **Operand store (os):** Write the result into memory or out to I/O.

Interrupts

- Virtually all computers provide a mechanism by which other modules (I/O, memory) may interrupt the normal processing of the processor
- Interrupts are provided primarily as a way to improve processing efficiency
- Common classes of interrupts
  - ❖Program
  - ❖Timer
  - ❖I/O
  - ❖Hardware Failure

Interrupts Cont.

- Program

Generated by some condition that occurs as a result of an instruction execution, such as arithmetic overflow, division by zero, attempt to execute an illegal machine instruction, or reference outside a user's allowed memory space.

- Timer

Generated by a timer within the processor. This allows the operating system to perform certain functions on a regular basis.

Interrupts Cont.

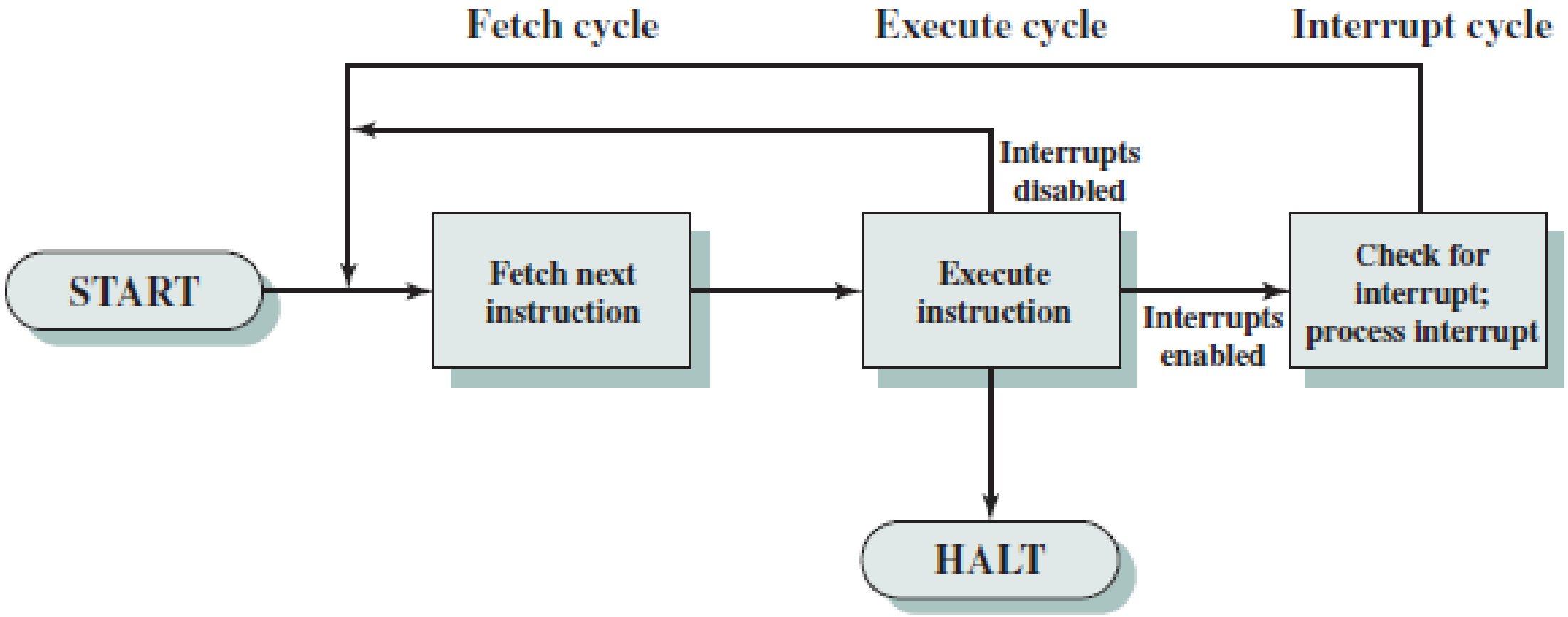
- I/O

Generated by an I/O controller, to signal normal completion of an operation, request service from the processor, or to signal a variety of error conditions.

- Hardware Failure

Generated by a failure such as power failure or memory parity error.

Interrupts and the instruction cycle



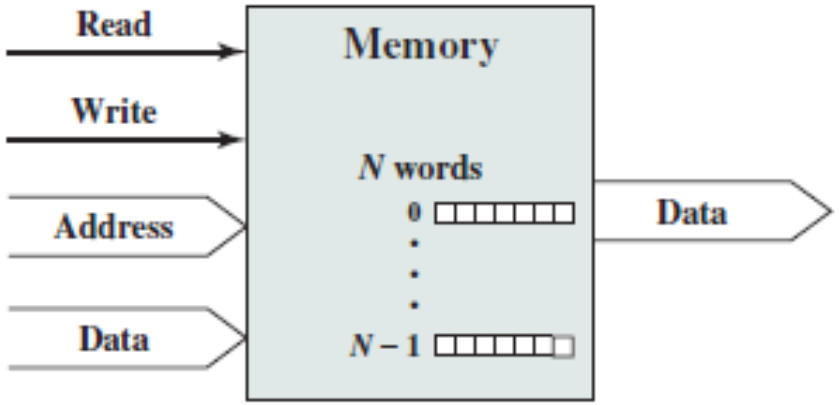
I/O Function

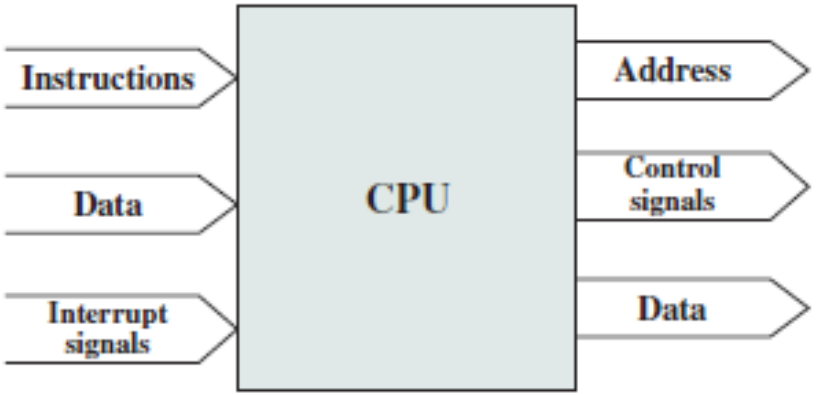
- An I/O module (e.g., a disk controller) can exchange data directly with the processor
- Just as the processor can initiate a read or write with memory, designating the address of a specific location, the processor can also read data from or write data to an I/O module
- In some cases, it is desirable to allow I/O exchanges to occur directly with memory
- This operation is known as Direct Memory Access (DMA)

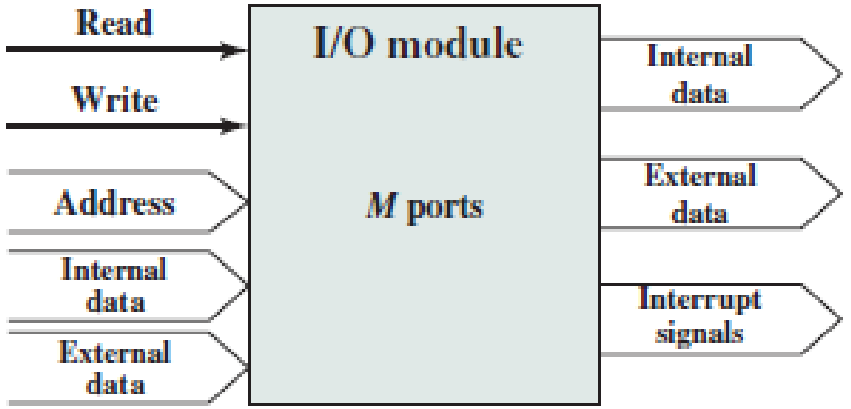
Interconnection Structures

- A computer consists of a set of components or modules of three basic types (processor, memory, I/O) that communicate with each other
- There must be paths for connecting the modules.
- The collection of paths connecting the various modules is called the interconnection structure
- The design of this structure will depend on the exchanges that must be made among modules.

Input and output for modules







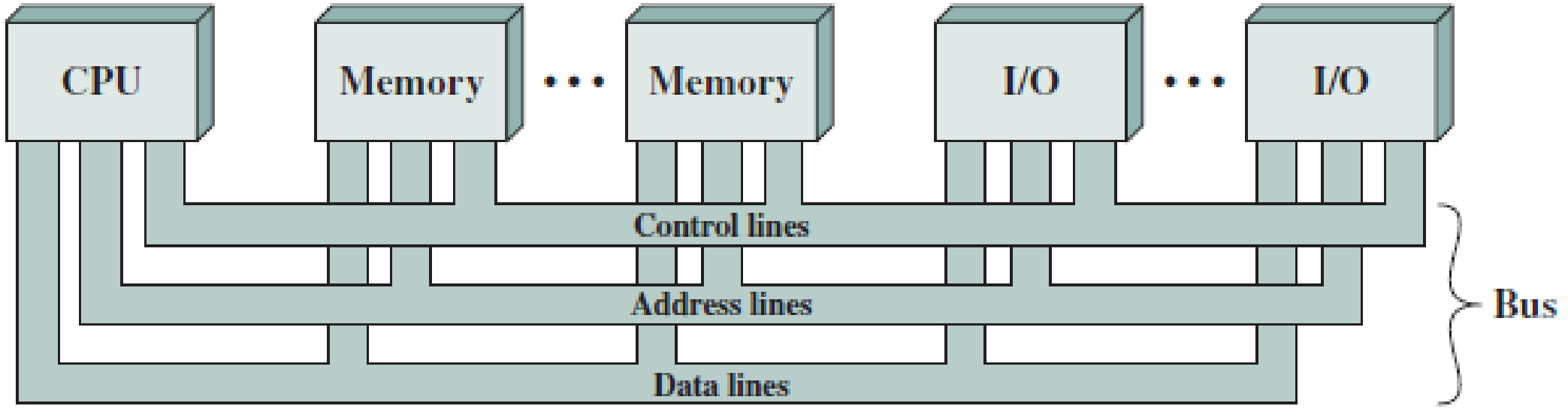
Types of transfers

- Memory to processor: The processor reads an instruction or a unit of data from memory.
- Processor to memory: The processor writes a unit of data to memory.
- I/O to processor: The processor reads data from an I/O device via an I/O module.
- Processor to I/O: The processor sends data to the I/O device.
- I/O to or from memory: For these two cases, an I/O module is allowed to exchange data directly with memory, without going through the processor, using direct memory access.

Bus Interconnection

- A bus is a communication pathway connecting two or more devices
- The bus is the dominant means of computer system component interconnection for decades
- A bus is shared transmission medium
- Multiple devices connect to the bus, and a signal transmitted by any one device is available for reception by all other devices attached to the bus
- If two devices transmit during the same time period, their signals will overlap and become garbled

Bus Interconnection Scheme



Functional Group of Bus Lines

- Data bus
  - ❖The data lines provide a path for moving data among system modules.
  - ❖The data bus may consist of 32, 64, 128, or even more separate lines referred to as the **width** of the data bus
  - ❖The number of lines determines how many bits can be transferred at a time.

Functional Group of Bus Lines Cont.

- Address lines
  - ❖The address lines are used to designate the source or destination of the data on the data bus
  - ❖Typically, the higher-order bits are used to select a particular module on the bus, and the lower-order bits select a memory location or I/O port within the module.
- Control lines
  - ❖The control lines are used to control the access to and the use of the data and address lines

Control lines Cont.

- Control signals transmit both command and timing information among system modules
- Timing signals indicate the validity of data and address information
- Command signals specify operations to be performed

Typical control lines include

- Memory write: causes data on the bus to be written into the addressed location
- Memory read: causes data from the addressed location to be placed on the bus
- I/O write: causes data on the bus to be output to the addressed I/O port.
- I/O read: causes data from the addressed I/O port to be placed on the bus.
- Transfer ACK: indicates that data have been accepted from or placed on the bus.

Typical control lines Cont.

- Bus request: indicates that a module needs to gain control of the bus.
- Bus grant: indicates that a requesting module has been granted control of the bus.
- Interrupt request: indicates that an interrupt is pending.
- Interrupt ACK: acknowledges that the pending interrupt has been recognized.
- Clock: is used to synchronize operations.
- Reset: initializes all modules

Computer Memory System

- Characteristics
  - ❖Location
  - ❖Capacity
  - ❖Unit of transfer
  - ❖Access method
  - ❖Performance
  - ❖Physical type
  - ❖Physical characteristics
  - ❖Organisation

Location

- The term location refers to whether memory is internal or external to the computer
  - ❖Internal: the main memory, registers and cache all are kind of internal memories
  - ❖External: peripheral storage device such as disk and tape. They are accessible to CPU via device controller.

Capacity

- Word size
  - ❖ For internal memory, this is expressed in bytes or words. The common word length is 8, 16, 32 or 64 bits
  - ❖The word is the natural unit of organisation, and is equal to the number of bits to represent instruction (some times there is exceptions for that).
- Number of words or Bytes

Unit of Transfer

- Internal
  - ❖For internal memory, the unit of transfer is equal to the number of electrical lines into and out of the memory module
  - ❖This may be equal to the word length, but is often larger, such as 64, 128, or 256 bits
- External
  - ❖Usually a block which is much larger than a word

Access Methods

- Sequential
  - ❖Start at the beginning and read through in order
  - ❖Access time depends on location of data and previous location
  - ❖e.g. tape
- Direct
  - ❖Individual blocks have unique address
  - ❖Access is by jumping to vicinity plus sequential search
  - ❖Access time depends on location and previous location
  - ❖e.g. disk

Access Methods Cont.

- Random
  - ❖Individual addresses identify locations exactly
  - ❖Access time is independent of location or previous access
  - ❖e.g. RAM
- Associative
  - ❖Data is located by a comparison with contents of a portion of the store
  - ❖Access time is independent of location or previous access
  - ❖e.g. cache

Performance

- Access time
  - ❖Time between presenting the address and getting the valid data
- Memory Cycle time
  - ❖Time may be required for the memory to “recover” before next access.
  - ❖Cycle time is access + recovery
- Transfer Rate
  - ❖Rate at which data can be moved

Transfer Rate Cont.

- Random Access
  - ❖Transfer rate=1/(cycle time).
- Non Random Access
- $T_n = T_A + \frac{n}{R}$

  Where

  T_n = Average time to read or write n bits

  T_A = Average access time

  n = Number of bits

  R = Transfer rate, in bits per second (bps)

Physical Types

- Semiconductor
  - ❖RAM
- Magnetic
  - ❖Disk & Tape
- Optical
  - ❖CD & DVD
- Others
  - ❖Bubble
  - ❖Hologram

Physical Characteristics

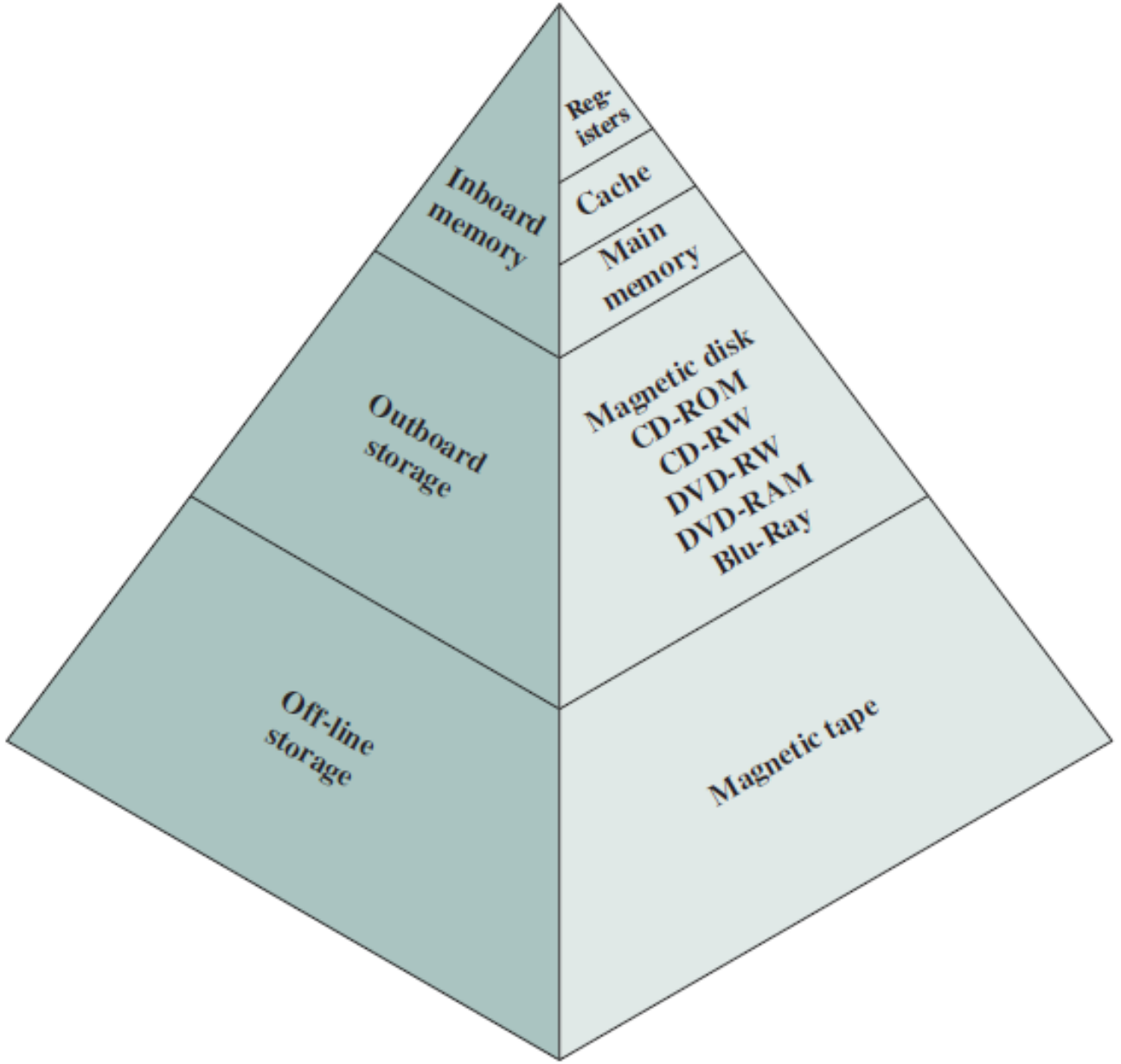
- Volatile/Non volatile
  - ❖In a volatile memory, information decays naturally or is lost when electrical power is switched off
  - ❖nonvolatile memory, information once recorded remains without deterioration until deliberately changed
- Nonerasable memory
  - ❖Cannot be altered, except by destroying the storage unit

The Memory Hierarchy

- The design constraints on a computer's memory can be summed up by three questions:
  - ❖How much?
  - ❖How fast?
  - ❖How expensive?

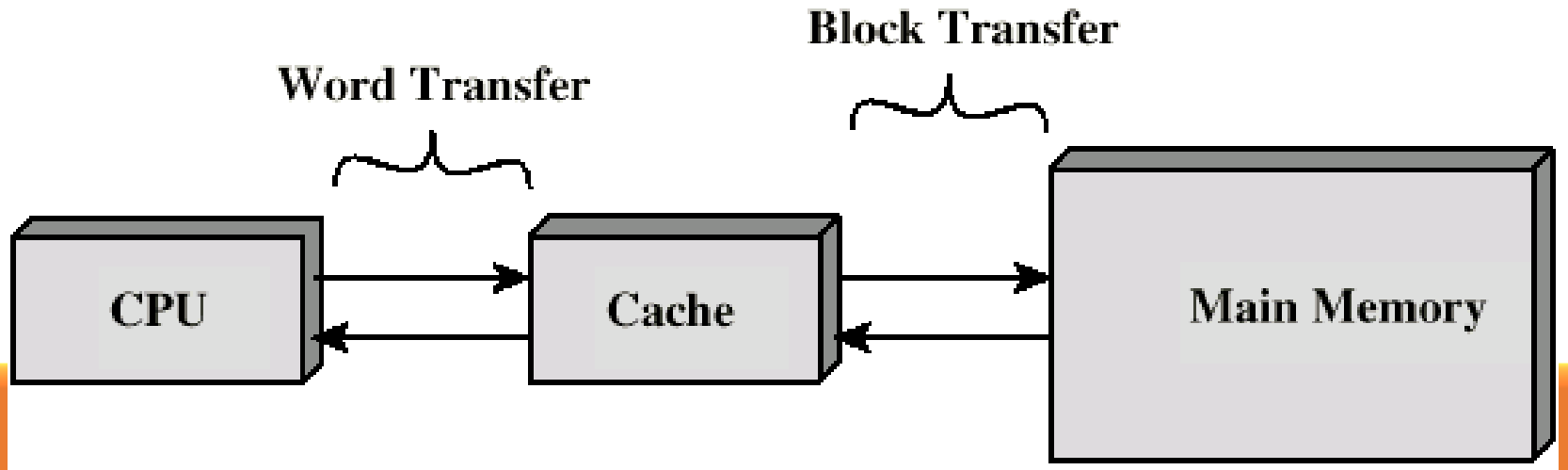
Hierarchy List

- Registers
- L1 Cache
- L2 Cache
- Main memory
- Disk cache
- Disk
- Optical
- Tape


Reg-
isters
Cache
Main
memory
Inboard
memory
Outboard
storage
Magnetic disk
CD-ROM
CD-RW
DVD-RW
DVD-RAM
Blu-Ray
Off-line
storage
Magnetic tape

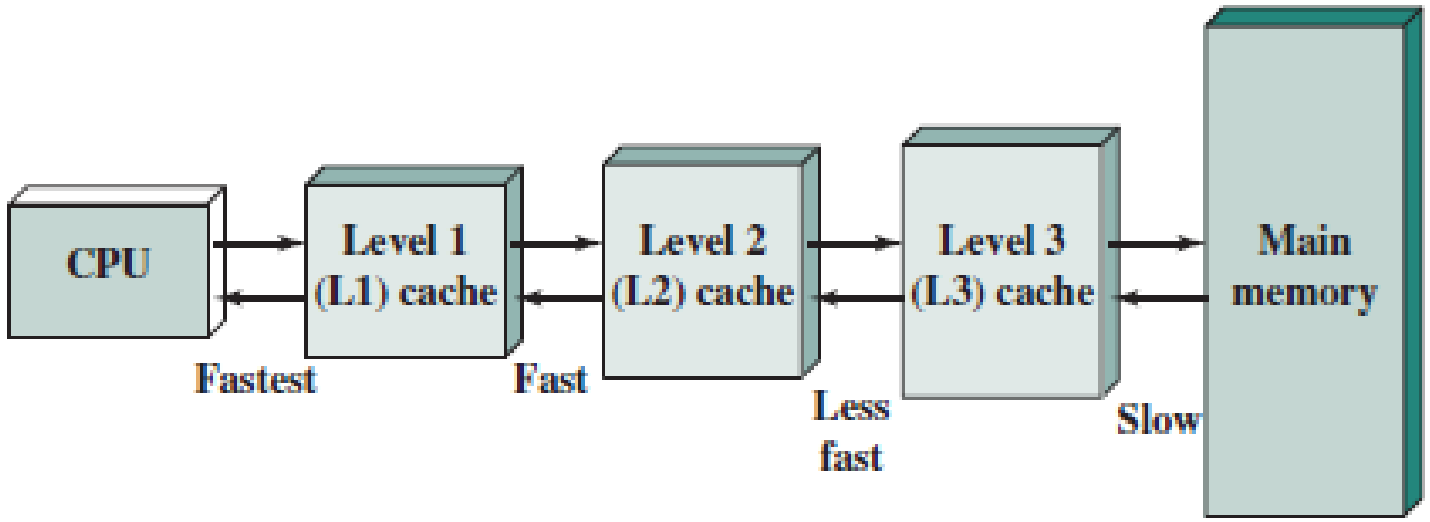
Cache

- Small amount of fast memory to improve the performance with cheap price.
- Sits between normal main memory and CPU to keep part of the data and instructions in the main memory.
- May be located on CPU chip or module



Cache Cont.

- Three-level cache organization



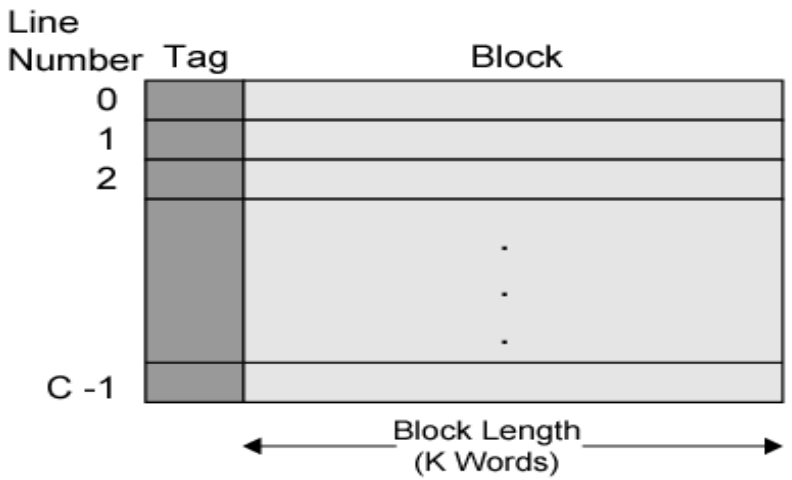
Cache Cont.

- Hit: when the processor fetch word and find it in the cache, otherwise Miss occurs.
- If miss occurs the block of the required word (consists of fixed number of words) is read from main memory and saved in the cache.
- Because of the locality principle, it likely to find the next referenced data or instructions in the cache as result of binging the whole block rather than the required word.

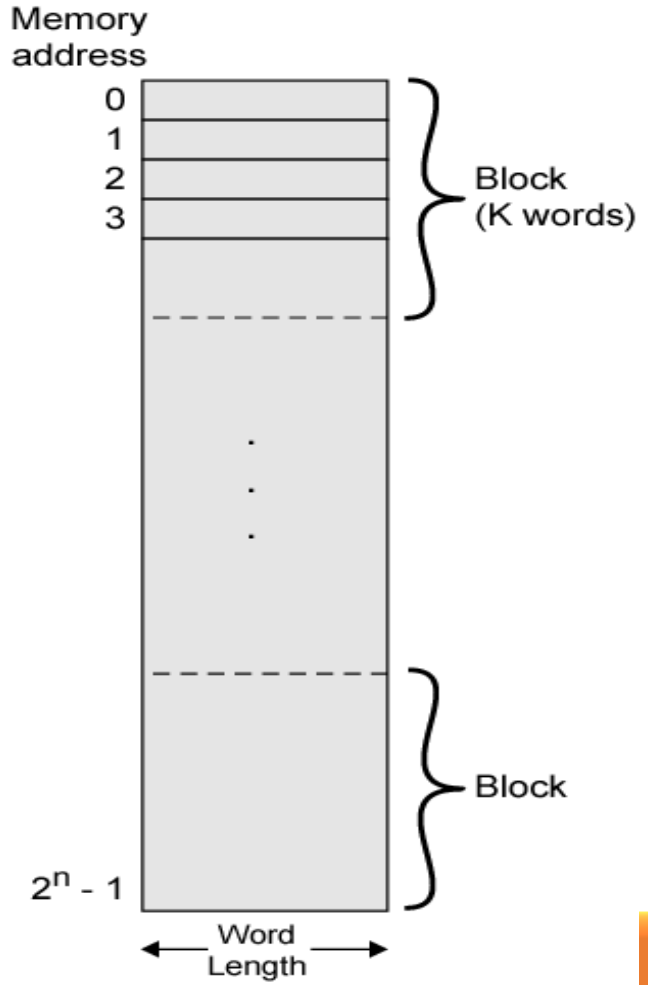
Cache Cont.

- Memory of size 2^n will have n address lines.
- The main memory is divided into blocks ,each block consists of k words (i.e. the number of blocks $M = 2^n/k$.
- The cache consists of lines C , each has the same size as block and $C << M$.
- At any time the cache will holds some blocks which are involved with processor work at that time.
- As the number of C is much less than the number of blocks, each line will have tag to indicate the block it holds.

Cache/Main Memory Structure



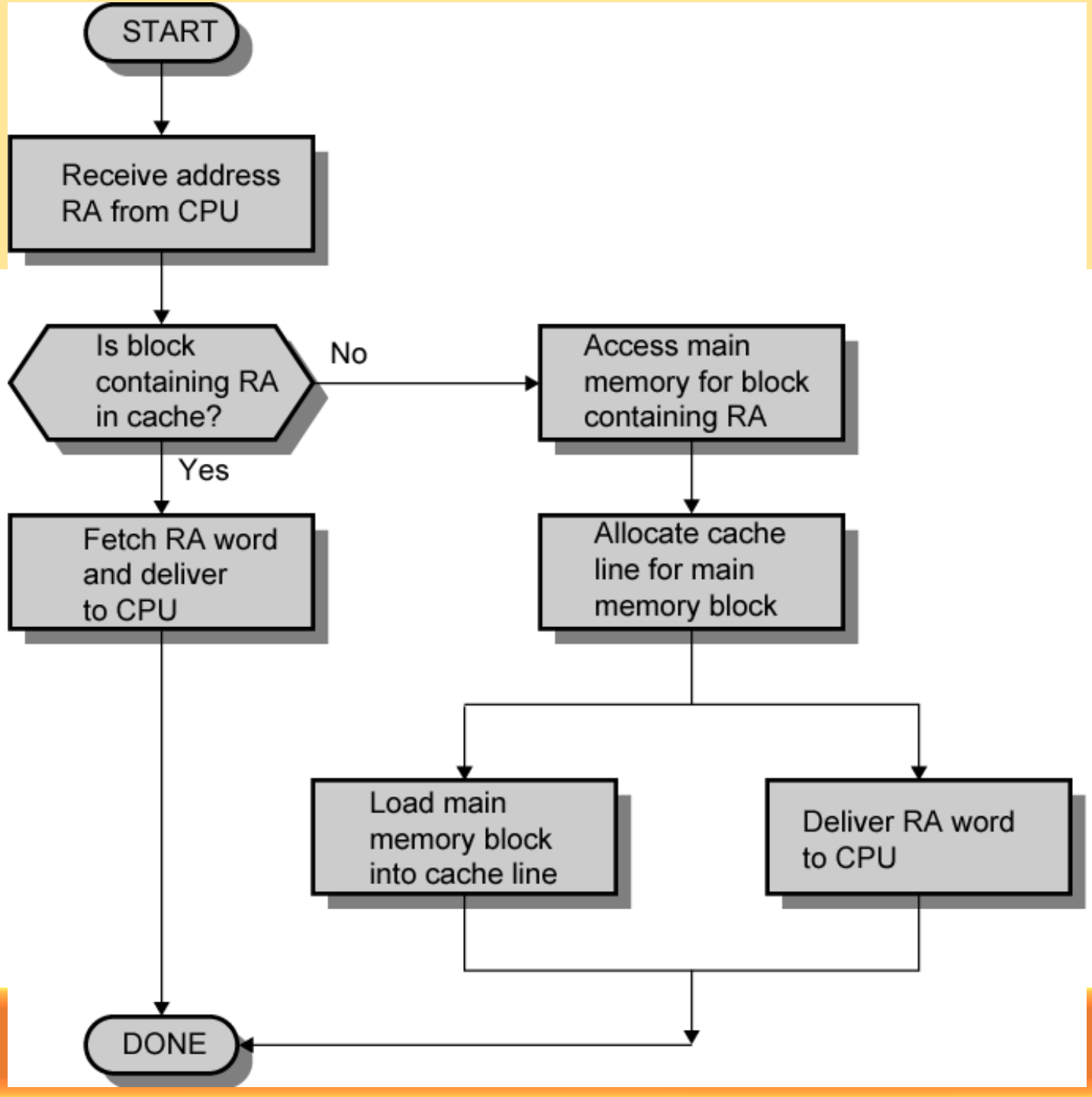
(a) Cache



(b) Main memory

Cache operation

- CPU requests contents of memory location
- Check cache for this data
- If present, get from cache (fast)
- If not present, read required block from main memory to cache
- Then deliver from cache to CPU
- Cache includes tags to identify which block of main memory is in each cache slot


START
Receive address RA from CPU
Is block containing RA in cache?
No
Yes
Access main memory for block containing RA
Fetch RA word and deliver to CPU
Allocate cache line for main memory block
Load main memory block into cache line
Deliver RA word to CPU
DONE

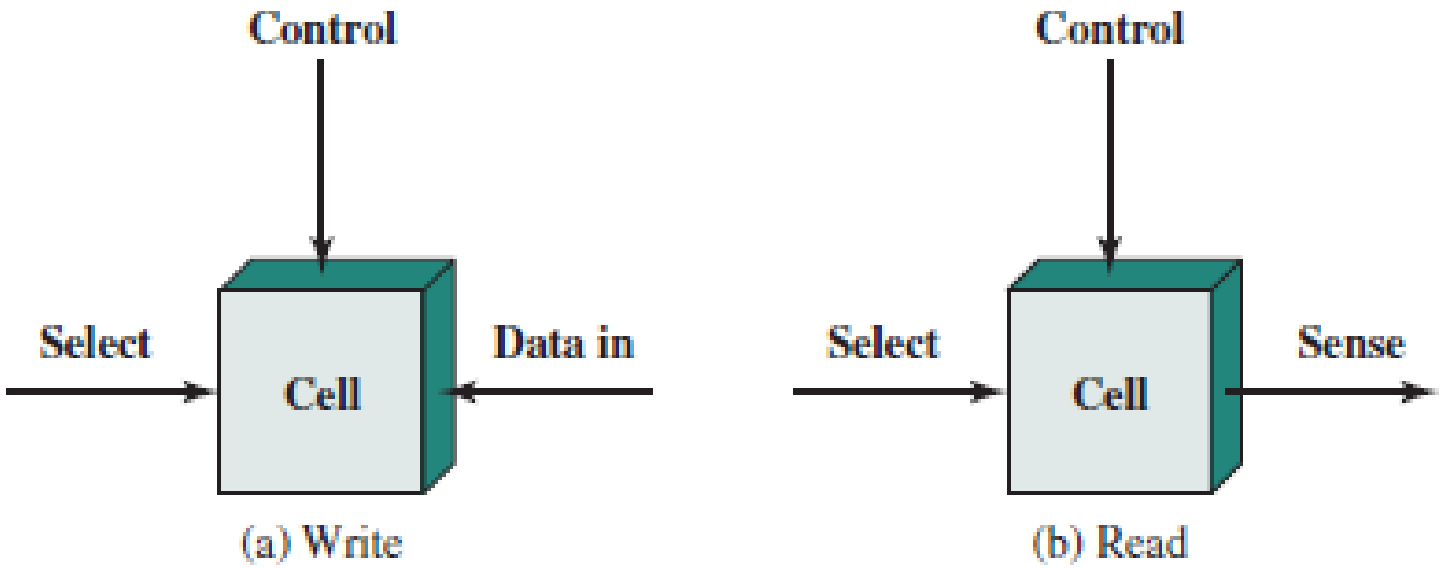
Cache Design Issue

- Size
- Mapping Function
- Replacement Algorithm
- Write Policy
- Block Size
- Number of Caches

Internal Memory

- The basic element of a semiconductor memory is the memory cell
- Although a variety of electronic technologies are used, all semiconductor memory cells share certain properties
  - ❖They exhibit two stable (or semis table) states, which can be used to represent binary 1 and 0.
  - ❖They are capable of being written into (at least once), to set the state.
  - ❖They are capable of being read to sense the state.

Memory Cell Operation



Semiconductor Memory Types

Memory Type	Category	Erasure	Write Mechanism	Volatility
Random-access memory (RAM)	Read-write memory	Electrically, byte-level	Electrically	Volatile
Read-only memory (ROM)	Read-only memory	Not possible	Masks	Nonvolatile
Programmable ROM (PROM)			Electrically	
Erasable PROM (EPROM)	Read-mostly memory	UV light, chip-level		
Electrically Erasable PROM (EEPROM)		Electrically, byte-level		
Flash memory		Electrically, block-level		

DRAM and SRAM

- RAM technology is divided into two technologies:
  - ❖Dynamic
  - ❖Static.
- As capacitors have a natural tendency to discharge, Dynamic RAMs require periodic charge refreshing to maintain data storage
- A Static RAM will hold its data as long as power is supplied to it

SRAM versus DRAM

- Both static and dynamic RAMs are volatile
- A dynamic memory cell is simpler and smaller than a static memory cell
- DRAM is less expensive than a corresponding SRAM
- DRAMs tend to be favored for large memory requirements
- SRAMs are somewhat faster than DRAMs
- SRAM is used for cache memory (both on and off chip), and DRAM is used for main memory

Types of ROM

- Read-Only Memory (ROM) contains a permanent pattern of data that cannot be changed.
- Other types of ROM
- Programmable ROM (PROM)
- Read-mostly memory
  - ❖Erasable programmable read-only memory (EPROM)
  - ❖Electrically erasable programmable read-only memory (EEPROM)